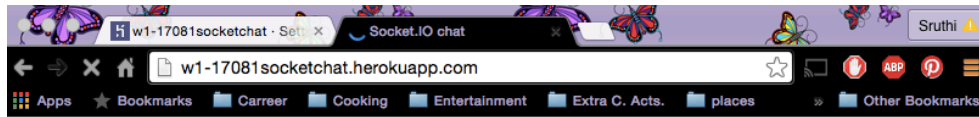

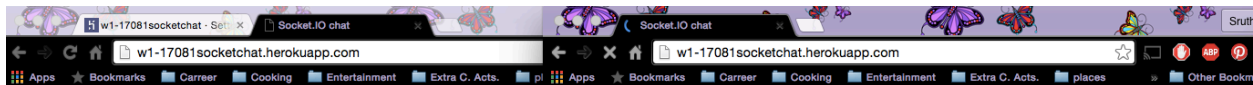



Chat example using socket.io

<http://w1-17081socketchat.herokuapp.com/>

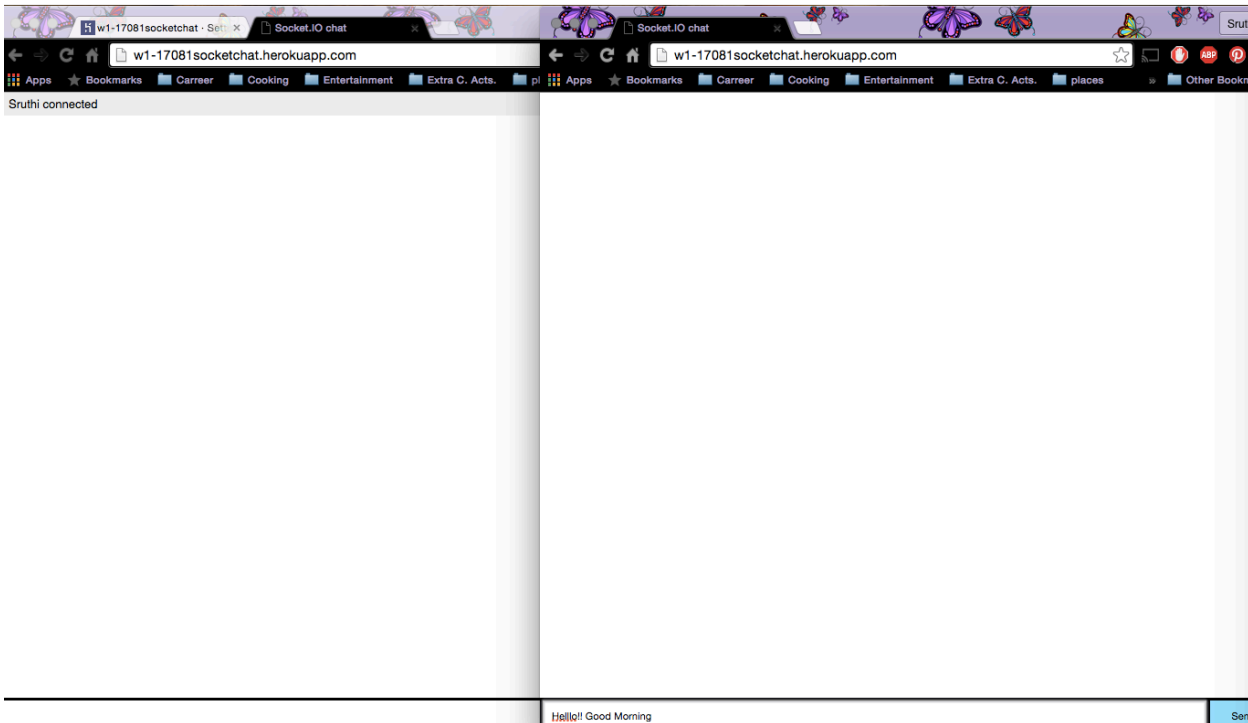
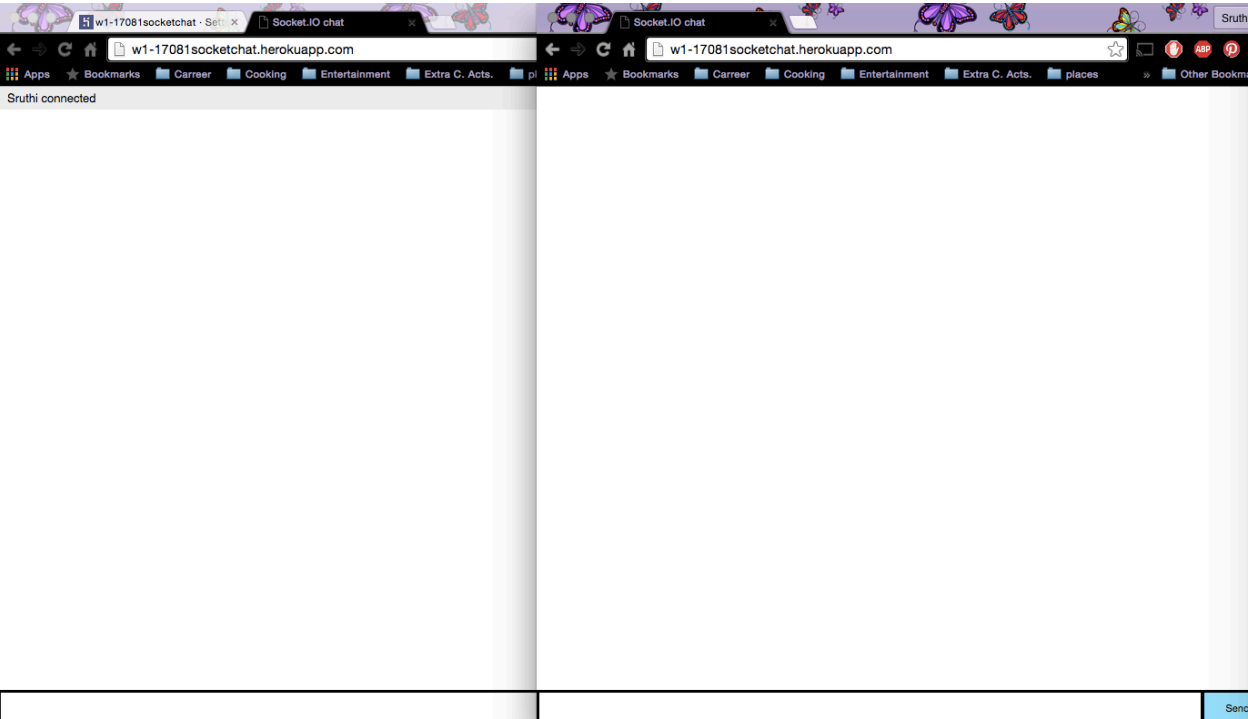


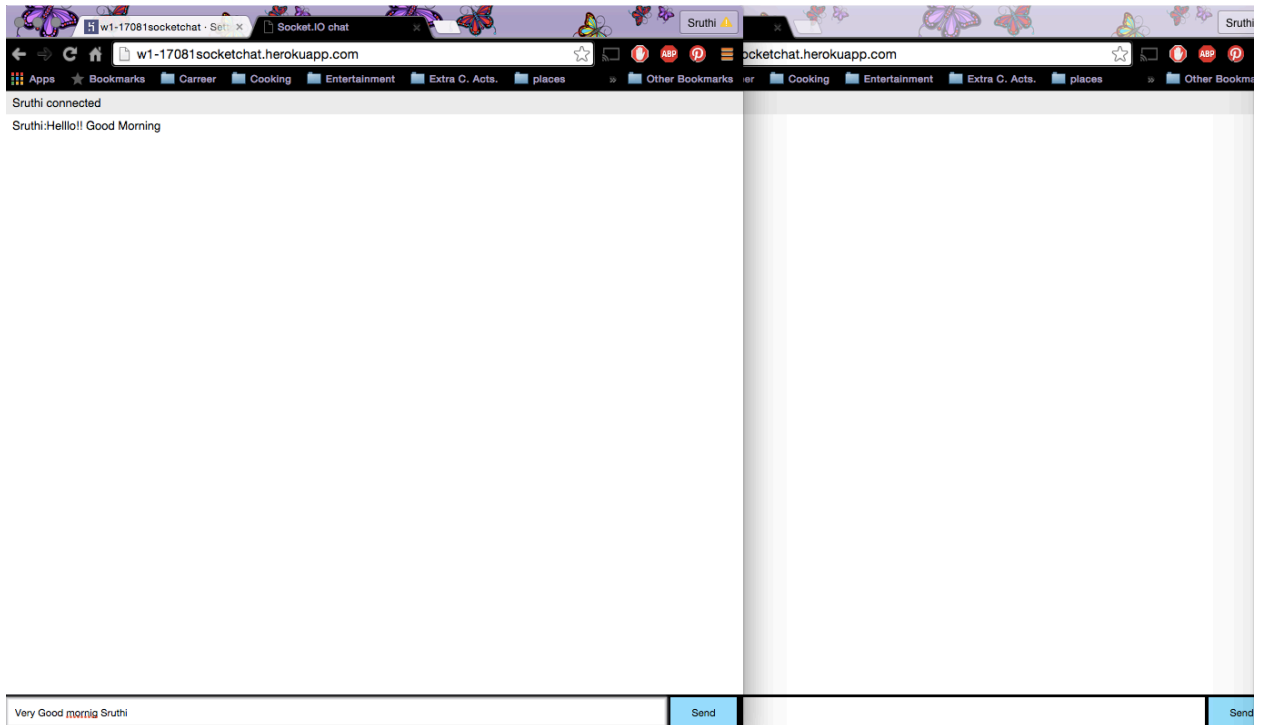
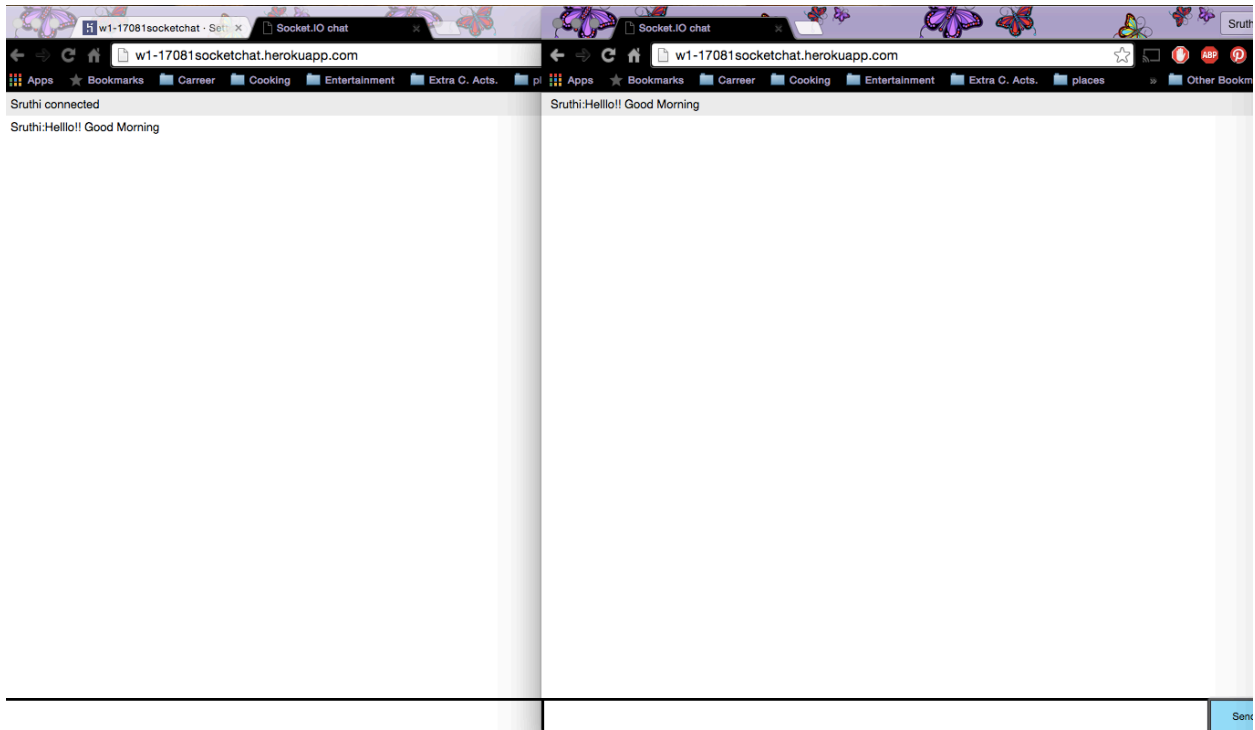
 **w1-17081socketchat.herokuapp.com says:**
Enter your name?

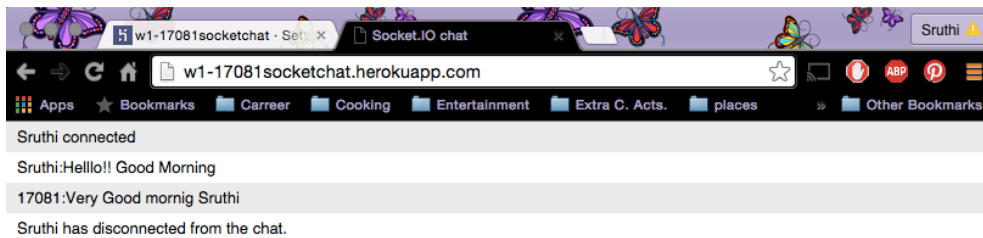
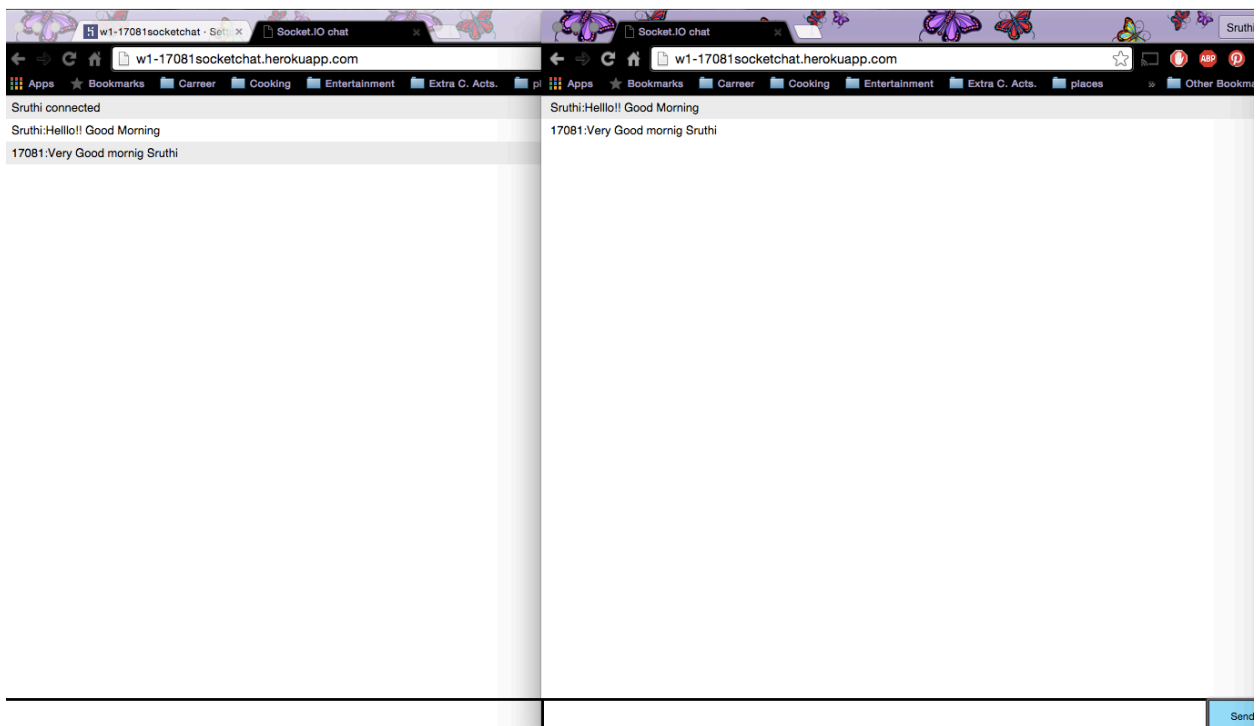


 **w1-17081socketchat.herokuapp.com says:**
Enter your name?
☐ Prevent this page from creating additional dialogs.

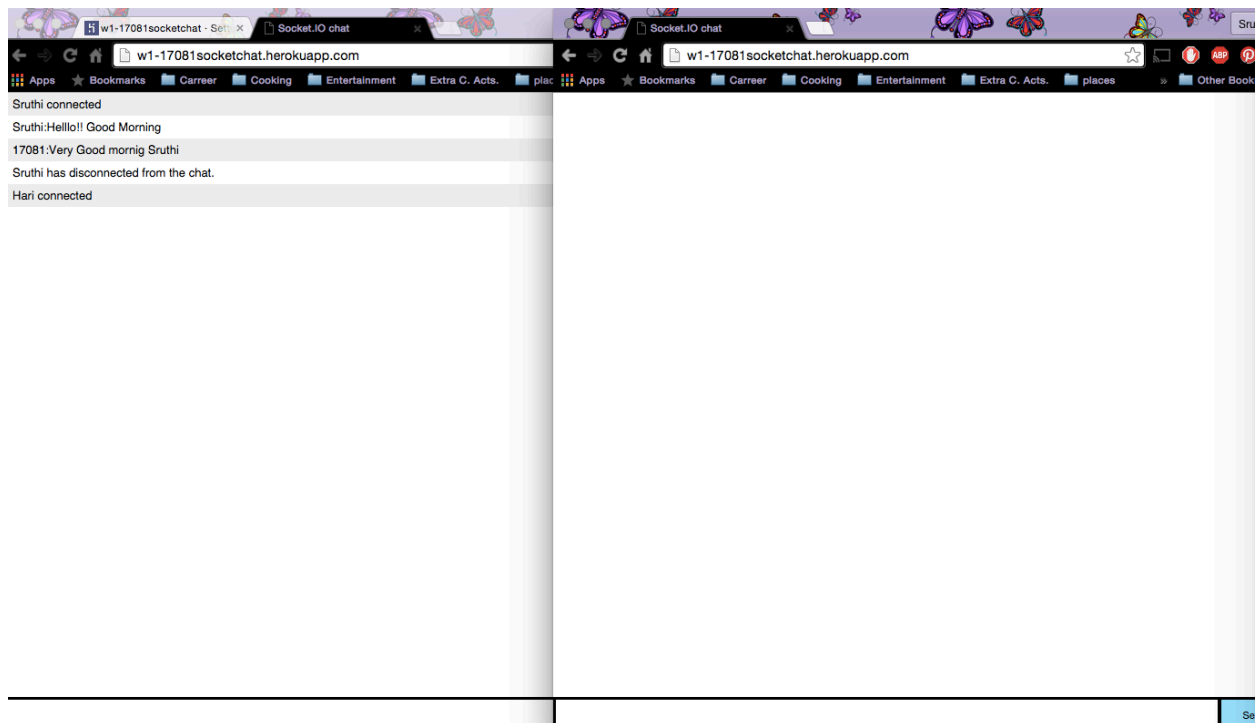
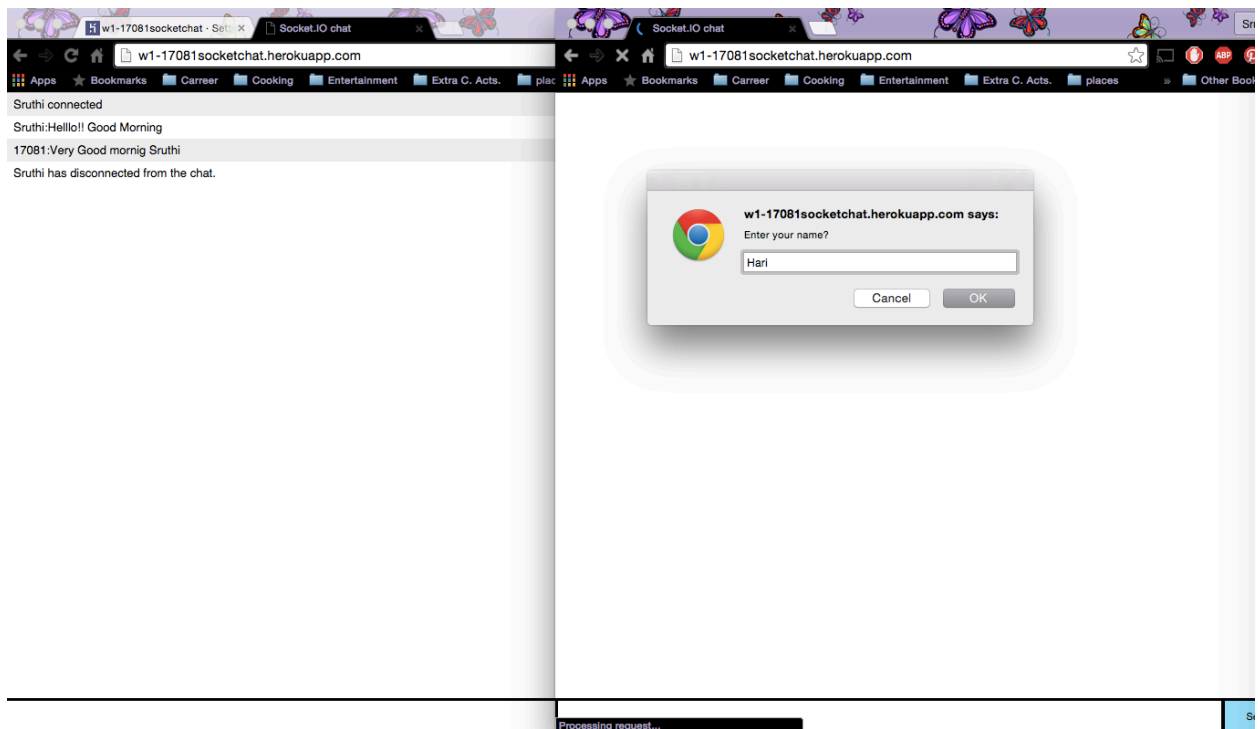
Send

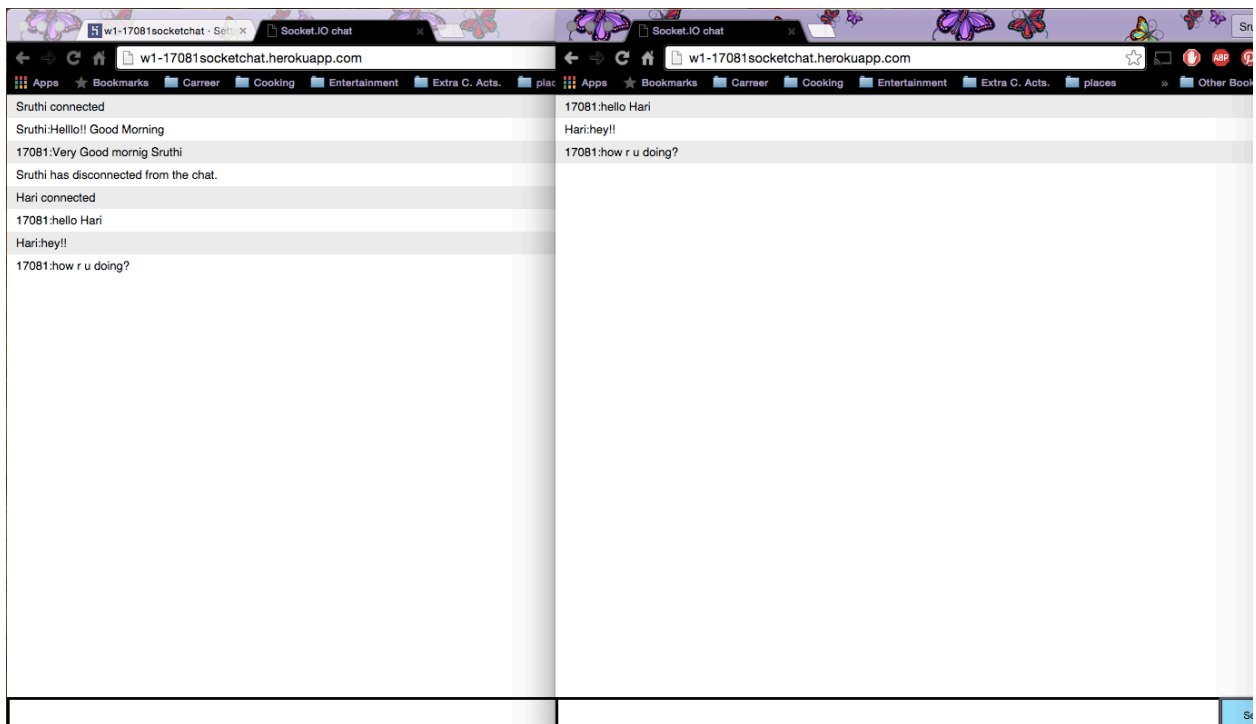
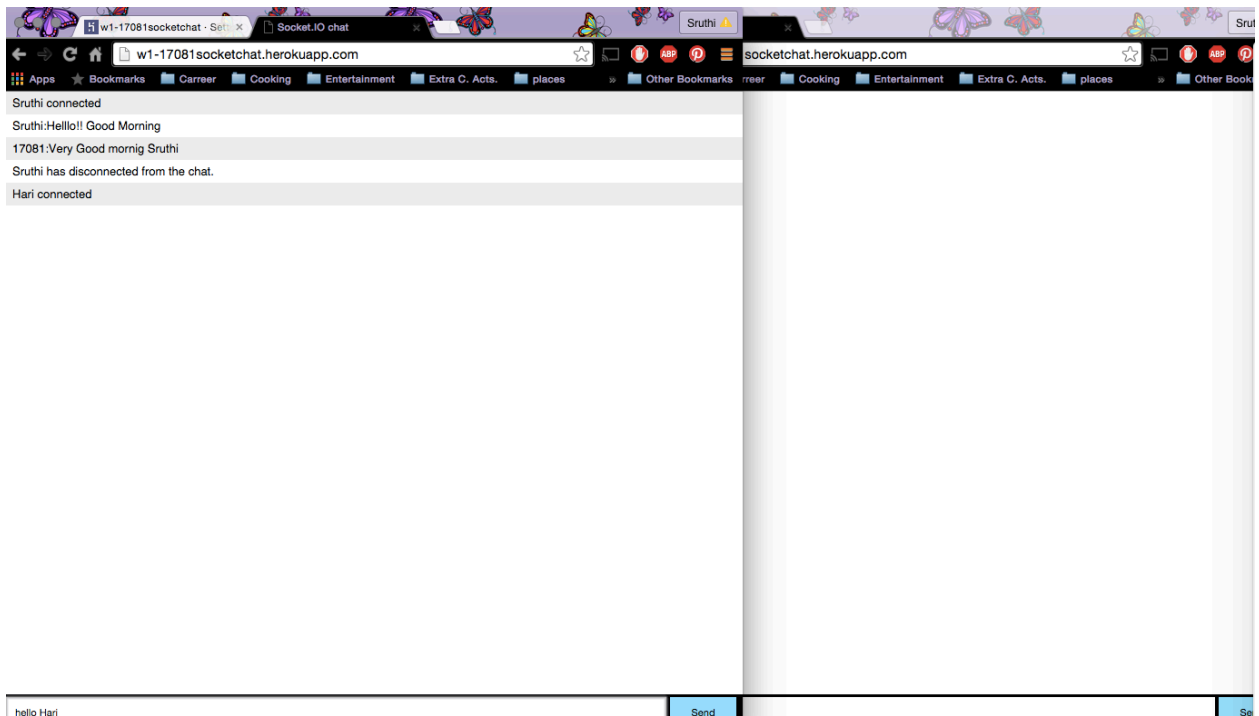


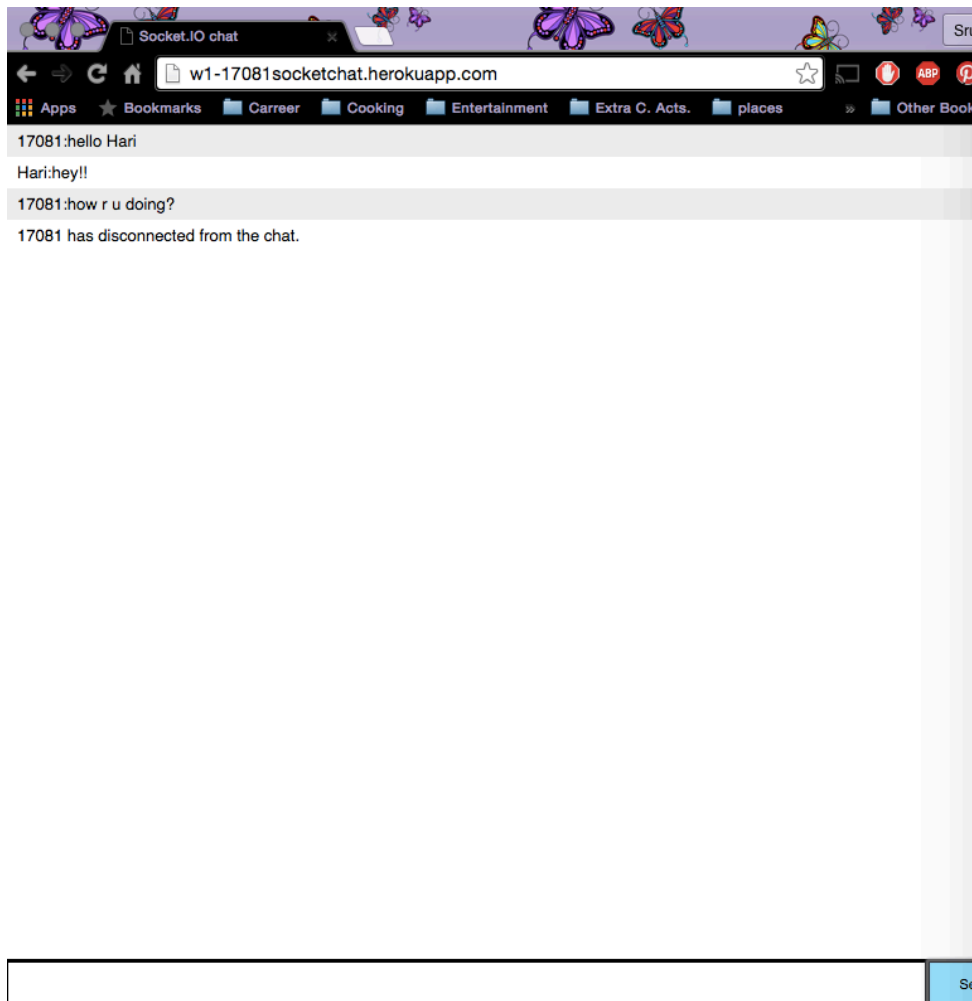




Send







```
index.html
UNREGISTERED

1 <!doctype html>
2 <html>
3 <head>
4 <title>Socket.IO chat</title>
5 <style>
6 * { margin: 0; padding: 0; box-sizing: border-box; }
7 body { font: 13px Helvetica, Arial; }
8 form { background: #000; padding: 3px; position: fixed; bottom: 0; width: 100%; }
9 form input { border: 0; padding: 10px; width: 90%; margin-right: .5%; }
10 form button { width: 9%; background: rgb(130, 224, 255); border: none; padding: 10px; }
11 #messages { list-style-type: none; margin: 0; padding: 0; }
12 #messages li { padding: 5px 10px; }
13 #messages li:nth-child(odd) { background: #eee; }
14 </style>
15 </head>
16 <body>
17 <ul id="messages"></ul>
18 <form action="">
19 <input id="m" autocomplete="off" /><button>Send</button>
20 </form>
21 <script src="/socket.io/socket.io.js"></script>
22 <script src="http://code.jquery.com/jquery-1.11.1.js"></script>
23
24
25 <script>
26 var socket = io();
27 var name = prompt('Enter your name?');
28 socket.on('connect', function(msg){
29   console.log(name);
30   while(name == 'null'){
31     name = prompt('Enter your name?')
32   }
33   socket.emit('join', name);
34 });
35
36 $('form').submit(function(){
37   socket.emit('chat message', $('#m').val());
38   $('#m').val('');
39   return false;
40 });
41
42 socket.on('chat message', function(msg){
43   $('#messages').append('<li>').text(msg);
44 });
45
46 </script>
47 </body>
```

Line 1, Column 1 Spaces: 2 HTML

```
server.js
UNREGISTERED

1 var app = require('express')();
2 var http = require('http').Server(app);
3 var io = require('socket.io')(http);
4
5 var port = process.env.PORT || 3000;
6
7
8 app.get('/', function(req, res){
9   res.sendFile(__dirname + '/index.html');
10 });
11
12 io.on('connection', function(socket){
13
14   socket.on('chat message', function(msg){
15     console.log('message: ' + msg);
16     io.emit('chat message', socket.name + ':' + msg);
17   });
18
19   socket.on('join', function(name){
20     socket.name = name;
21     socket.broadcast.emit('chat message', name + ' connected');
22     console.log('user connected' + name);
23   });
24
25   socket.on('disconnect', function(){
26     if(socket.name !== undefined){
27       console.log('user disconnected');
28       socket.broadcast.emit('chat message', socket.name + ' has disconnected from the chat. ');
29     }
30   });
31 });
32
33 http.listen(port, function(){
34   console.log('listening on *:3000');
35 });
36
37
38
39
40
41
42
```

Line 1, Column 1 Spaces: 2 JavaScript